CORE
Metadata, citation and similar papers at core.ac.uk
Provided by Brunel University Research Archive

Applying adaptive test cases to nondeterministic implementations

R. M. Hierons [a]

[a] *School of Information Systems, Computing and Mathematics, Brunel University, Uxbridge, Middlesex, UB8 3PH, United Kingdom*

Key words: State-based testing; Test Execution; Adaptive test cases; Nondeterministic implementation; Minimising Cost.

1 Introduction

The testing of a state-based system involves the application of sequences of inputs and the observation of the resultant input/output sequences (*traces*). These traces can result from preset input sequences or adaptive test cases in which the choice of the next input depends on the trace that has observed up to that input. Adaptive test cases are used in a number of areas including protocol conformance testing (see, for example, [1,6,9]) and adaptivity forms the basis of the standardised test language TTCN (see, for example, [2]).

Suppose that we apply adaptive test case γ to the system under test (SUT) and observe the trace $\bar{\sigma}$. If the SUT is deterministic and we apply γ again, after resetting the SUT, then we will observe $\bar{\sigma}$ again. Further, if we have another adaptive test case γ' where a prefix $\bar{\sigma}'$ of $\bar{\sigma}$ is a possible response to γ' then we know that the application of γ' must lead to $\bar{\sigma}'$. Thus, for a deterministic SUT the response of the SUT to an adaptive test case γ' might be deduced from the response of the SUT to another adaptive test case γ [4]. This observation can be used to reduce the cost of testing: we only apply adaptive test case γ' if we cannot deduce the response to γ' from the set of observations [4].

While many systems are deterministic, nondeterminism is becoming increasingly common. Nondeterminism in the SUT is typically a consequence of limits in the ability to observe the SUT. For example, it could be a result of information hiding, real time properties, or of different possible interleavings in a concurrent system (see, for example, [10]). This paper investigates the case where the SUT is nondeterministic. We consider the situation in which a set O of traces has been observed in testing and we are considering applying an adaptive test case γ. In general we cannot expect to be able to deduce the response of a nondeterministic SUT to an adaptive test case γ since there may be more than one possible response. Instead we consider the question of how we can decide whether the application of γ could lead to a trace that has not been observed. A solution to this would allow us to reduce the cost of testing: if all possible responses of the SUT to γ have already been observed then we do not have to apply γ in testing and thus reduce the cost of test execution.

This paper considers three cases. Section 3 considers the case where we can apply a fairness assumption. Section 4 weakens this assumption to us having a lower bound p on the probability of observing alternative responses of the SUT to any input and in any state. Section 5 then considers the general case.

2 Preliminaries

2.1 Sequences

Throughout this paper X and Y denote the finite input and output domains of the SUT. For a set A, A^* denotes the set of finite sequences of elements of A including the empty sequence ϵ. A variable representing a sequence will have a bar over its name (for example, $\bar{x}$). The concatonation of sequences $\bar{c}$ and $\bar{d}$ is represented by $\bar{c}\bar{d}$. Given a set B of sequences from A^*, $Pre(B) = \{\bar{b}' | \exists \bar{b} \in B, \bar{b}'' \in A^*. \bar{b} = \bar{b}'\bar{b}''\}$ denotes the set of prefixes of sequences from B.

We assume that the SUT is a black box state-based system whose functional behaviour is being tested. In testing we thus observe traces of the form $\langle x_1/y_1, \ldots, x_k/y_k \rangle \in (X/Y)^*$ where $\bar{x} = x_1, \ldots, x_k \in X$ and $\bar{y} = y_1, \ldots, y_k \in Y$ and we also represent such a trace as $\bar{x}/\bar{y}$. A preset test sequence is an element of X^*. If a set of test sequences, or adaptive test cases, is applied then the SUT is returned to its initial state after each test using a test postamble [5]. This ensures that each adaptive test case or test sequence is applied in the same state of the SUT. The postamble could involve the system being reset or switched off and then on again, an adaptive process, or a sequence of inputs. Throughout this paper, given an SUT I and an input sequence $\bar{x}$, $I(\bar{x})$ denotes the set of possible output sequences I can produce in response to $\bar{x}$.

2.2 Adaptive test cases

In this paper $\mathcal{T}$ denotes the set of all adaptive test cases that have (finite) input domain X and (finite) output domain Y. The set $\mathcal{T}$ can thus be recursively

BURA

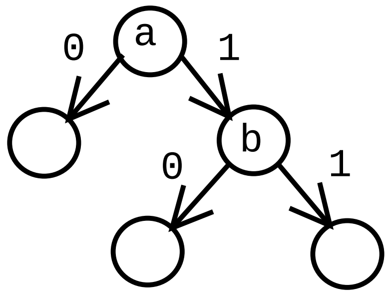

Fig. 1. An adaptive test case

defined in the following manner [4].

Definition 1 *Each element $\gamma \in \mathcal{T}$ is either null or a pair (x, f) in which $x \in X$ and f is a function from Y to $\mathcal{T}$.*

We apply adaptive test case γ in the following way. If $\gamma = null$ then we terminate. Otherwise $\gamma = (x, f)$ and we apply x, observe an output y, and then apply adaptive test case $f(y)$ [1]. An adaptive test case can be represented by a tree such as that shown in Figure 1 and we assume that this representation is finite. Figure 1 represents an adaptive test case in which we apply input a, terminating if the output is 0. If the output is 1 then we apply input b and whatever the response to b after a, the adaptive test case then terminates. This adaptive test case can be defined by: $\gamma = (a, f)$, $f(0) = null$, $f(1) = (b, f')$, $f'(0) = null$, and $f'(1) = null$.

Given an adaptive test case γ, $IO(\gamma)$ denotes the set of traces that can be observed using γ [4]. For example, the adaptive test case γ given in Figure 1 has: $IO(\gamma) = \{\langle a/0\rangle, \langle a/1, b/0\rangle, \langle a/1, b/1\rangle\}$. Since every adaptive test case is represented by a finite tree, $IO(\gamma)$ is finite and every element of $IO(\gamma)$ is finite. Naturally, when γ is applied to an actual implementation this SUT may only be able to produce a subset of $IO(\gamma)$ in response to γ. The length of $\gamma \in \mathcal{T}$, $length(\gamma)$, is the length of the longest trace in $IO(\gamma)$ [4]. Given $\gamma \in \mathcal{T}$, the size $|\gamma|$ of γ is the number of nodes in the tree that represents γ [4]. Given $\bar{\sigma} \in Pre(IO(\gamma))$, $\gamma_{\bar{\sigma}}$ denotes the adaptive test case obtained if we have been applying γ and have observed trace $\bar{\sigma}$; this corresponds to the node of the tree representing γ that is reached from the root by a path with label $\bar{\sigma}$.

Throughout this paper O is the set of traces that have already been observed. Previous work has shown that sometimes we can deduce the response of a deterministic SUT to some adaptive test case γ on the basis of O [4]. In this paper we generalise this to a nondeterministic SUT and consider the problem of deciding whether the SUT can produce any response to adaptive test case γ that is not in $Pre(O)$: if the SUT cannot produce a trace that is not in

[1] We could allow f to be a partial function, where we terminate if f is applied to some $y \notin dom\ f$, but to simplify the exposition we only consider total functions.

BURA

$Pre(O)$ then testing with γ can provide no additional information [2].

3 Using the fairness assumption

In this section we assume that we have the following fairness assumption [8]: we have an integer k such that if we apply any input sequence $\bar{x}$ to the SUT k times then all possible responses of the SUT to $\bar{x}$ are observed. We also assume that each test sequence or adaptive test case [3] has been applied at least k times. We weaken this assumption in the following sections. We let Φ denote the set of possible SUTs for which we could have observed O in testing while satisfying this fairness assumption.

We define a recursive function det^1_f that takes the trace $\bar{\sigma}$ that has been observed so far in an adaptive test case, the current node (either $null$ or a pair (x, f)), and the set $O_{\bar{\sigma}}$ of traces that follow $\bar{\sigma}$ in O and decides whether the possible responses of the SUT to (x, f) after $\bar{\sigma}$ are in $O_{\bar{\sigma}}$.

Definition 2 *The function det^1_f is defined by the following rules in which $\bar{\sigma}$ is a trace, (x, f) is an adaptive test case, and $O_{\bar{\sigma}}$ is a set of traces.*

$$
\begin{aligned}
det^1_f(\bar{\sigma}, null, O_{\bar{\sigma}}) &= true\\
det^1_f(\bar{\sigma}, (x, f), O_{\bar{\sigma}}) &= (\exists y \in Y.x/y \in Pre(O_{\bar{\sigma}})) \wedge\\
&\qquad (\forall y \in Y.(x/y \in Pre(O_{\bar{\sigma}})) \Rightarrow det^1_f(\bar{\sigma}x/y, f(y), O_{\bar{\sigma}x/y}))
\end{aligned}
$$

The base case is trivial: we know the response to $null$. In the recursive case, if we apply (x, f) after $\bar{\sigma}$ then we require that x has been applied after $\bar{\sigma}$ and for all $y \in Y$, either the SUT cannot produce $\bar{\sigma}x/y$ (this has not been observed in O and we have repeated the application of each adaptive test case at least k times) or the set of possible responses to $f(y)$ after $\bar{\sigma}x/y$ is in $O_{\bar{\sigma}x/y}$. We can now define function det_f that takes $\gamma \in \mathcal{T}$ and a set O of observed traces and decide whether O contains all possible responses of the SUT to γ.

Definition 3 *The function det_f is defined by $det_f(\gamma, O) = det^1_f(\epsilon, \gamma, O)$.*

Proposition 1 *Suppose that if we apply any $\bar{x} \in X^*$ to the SUT k times then all possible responses of the SUT to $\bar{x}$ will be observed and that the set O of traces have been observed in response to the application of a set of adaptive*

[2] We make an implicit assumption here that in testing we only observe traces.

[3] It has been shown that if the fairness assumption holds for input sequences then the corresponding fairness assumption for adaptive test cases must also hold [3].

BURA

test cases k times. Then the response of the SUT to adaptive test case γ is guaranteed to be contained in $Pre(O)$ if and only if $det_f(\gamma, O)$ is true.

Proof. First assume that $det_f(\gamma, O) = true$ and let I denote an element of Φ. Let O' denote the set of traces that can be produced by I in response to γ: we require to prove that $O' \subseteq Pre(O)$. Proof by induction on the length of traces in $Pre(O')$. The result trivially holds for the base case $null$. Inductive hypothesis: every trace of length j or less in $Pre(O')$ is also in $Pre(O)$. Let us suppose that $\bar{\sigma} = \bar{\sigma}_1 x/y \in Pre(O')$ has length $j+1$. It is sufficient to prove that $\bar{\sigma} \in Pre(O)$. By the inductive hypothesis $\bar{\sigma}_1 \in Pre(O)$. Thus, since $det_f(\gamma, O)$ holds we must have that $det^1_f(\bar{\sigma}_1, (x, f), O_{\bar{\sigma}_1}) = true$, $(x, f) = \gamma_{\bar{\sigma}_1}$. Thus, there exists some $y' \in Y$ such that $\bar{\sigma}_1 x/y' \in Pre(O)$ and thus in testing x has been input after $\bar{\sigma}_1$. Thus, by the fairness assumption, $\bar{\sigma}_1 x/y \in Pre(O)$ as required.

Now suppose that $det_f(\gamma, O)$ is false (and so $\gamma \neq null$). By definition there exists $\bar{x}/\bar{y} \in Pre(IO(\gamma))$ such that $\bar{x}/\bar{y} \in Pre(O)$, $(x, f) = \gamma_{\bar{x}/\bar{y}} \neq null$, and for all $y \in Y$ we have that $\bar{x}x/\bar{y}y \notin Pre(O)$. But from this we can deduce that the SUT I can produce $\bar{x}/\bar{y}$ but that in testing we have not followed $\bar{x}/\bar{y}$ by the input of x. Thus, there exists some $y \in Y$ such that $\bar{y}y \in I(\bar{x}x)$ and thus there is some response of the SUT to γ that is not contained in $Pre(O)$ as required. $\Box$

Proposition 2 *Given adaptive test case γ and a set O of traces, $det_f(\gamma, O)$ can be computed in time of $O(|Y||O|length(\gamma))$.*

Proof. Given a node γ' of γ, let the depth of γ' denote the length of the trace that labels the path from the root of γ to γ'. Let m denote an integer with $m < length(\gamma)$ and let $\gamma_1, \ldots, \gamma_l$ denote the nodes with depth m. Given node γ_i let $\bar{\sigma}_i$ denote the label of the path from the root of γ to the node γ_i. Clearly $\sum_{j=1}^{l} |O_{\bar{\sigma}_i}| \leq |O|$.

Given $y \in Y$ and $\gamma_i = (x, f)$, it takes $O(|O_{\bar{\sigma}_i}|)$ effort to produce $O_{\bar{\sigma}_i x/y}$. Thus, the overall effort to produce the $O_{\bar{\sigma}_i x/y}$ is of $O(\sum_{j=1}^{l} \sum_{y \in Y} |O_{\bar{\sigma}_j}|) = O(\sum_{j=1}^{l} |Y||O_{\bar{\sigma}_j}|)$ and so, since $\sum_{j=1}^{l} |O_{\gamma_j}| \leq |O|$, this is of $O(|Y||O|)$. The result follows from observing that this does not depend on m and so the overall time complexity is of $O(\sum_{m=1}^{length(\gamma)} |Y||O|)$ which equals $O(|Y||O|length(\gamma))$. $\Box$

Note that we do not claim that the algorithm has optimal time complexity; the key point is that it has low-order polynomial time complexity.

4 A bound on probabilities

In this section we assume that if the SUT can produce trace $\bar{\sigma}x/y$ and we have observed $\bar{\sigma}$ then the probability of receiving y in response to x is at least p and that this holds for all x, y, and $\bar{\sigma}$. This assumption allows us to provide confidence in the SUT not being able to produce trace $\bar{x}/\bar{y}$ if we have an adaptive test case γ with $\bar{x}/\bar{y} \in Pre(IO(\gamma))$ that has been applied a sufficient number of times, and we have not observed $\bar{x}/\bar{y}$. We can ask the following question: what is the probability that we do not observe $\bar{x}/\bar{y}$ if $\bar{x}/\bar{y}$ is a trace of the SUT and we have applied an adaptive test case γ with $\bar{x}/\bar{y} \in Pre(IO(\gamma))$ m times? By investigating this question, we can set m so that if we do not observe $\bar{x}/\bar{y}$ then we have a given confidence in the SUT not being able to produce $\bar{x}/\bar{y}$. The following is clear.

Proposition 3 *Suppose that $\bar{x}/\bar{y} \in Pre(IO(\gamma))$ and we apply γ to the SUT. If the SUT can produce $\bar{y}$ in response to $\bar{x}$ then the probability of observing $\bar{x}/\bar{y}$ is bounded below by p^j where $j = |\bar{x}|$.*

Suppose that the SUT can produce the trace $\bar{x}/\bar{y}$. The probability of not observing $\bar{x}/\bar{y}$ is at most $1-p^j$ and so the probability of not observing $\bar{x}/\bar{y}$ in m tests with γ is at most $(1-p^j)^m$. Given $0 < \delta < 1$ we can choose m such that the chance of failing to observe $\bar{x}/\bar{y} \in Pre(IO(\gamma))$ in m tests with γ is at most δ if the SUT can produce $\bar{x}/\bar{y}$. It is sufficient to choose some m with $\delta \geq (1-p^{length(\gamma)})^m$. Since log is a monotonically increasing function this is equivalent to:

$$\log \delta \geq \log((1-p^{length(\gamma)})^m) = m \log(1-p^{length(\gamma)})$$

Since $\log(1-p^{length(\gamma)}) < 0$, if we divide both sides of the above inequality by $\log(1-p^{length(\gamma)})$ we change $\geq$ to $\leq$. Thus we require m to be an integer that satisfies the following.

$$m \geq \frac{\log \delta}{\log(1-p^{length(\gamma)})}$$

Proposition 4 *Suppose that if the SUT can produce trace $\bar{\sigma}x/y$ then the probability of receiving y in response to x after $\bar{\sigma}$ is at least p and that this holds for all x, y, and $\bar{\sigma}$. Let Γ' denote a set of adaptive test cases and each adaptive test case $\gamma' \in \Gamma'$ is to be applied at least the following number of times:*

$$\frac{\log \delta}{\log(1-p^{length(\gamma')})}$$

Let E' denote a subset of $Pre(IO(\gamma))$, $\gamma \notin \Gamma'$, and suppose that for every sequence $\bar{x}/\bar{y} \in E'$ there exists $\gamma' \in \Gamma'$ such that $\bar{x}/\bar{y} \in Pre(IO(\gamma'))$. Then *the probability that none of the traces in E' will be observed in the application of the elements of Γ', if one or more of them can be produced by the SUT in response to γ, is at most $|E'|\delta$.*

Proof. Let $E' = \{\bar{x}_1/\bar{y}_1, \ldots, \bar{x}_\alpha/\bar{y}_\alpha\}$. Observe that the probability that none of the elements of E' are observed if one or more of them can be produced by the SUT is at most the sum over $1 \leq i \leq \alpha$ of the probability that $\bar{x}_i/\bar{y}_i$ is not observed if it can be produced by the SUT. This is bounded above by $|E'|\delta$ and the result thus follows. □

We can now give a condition under which we can have confidence at least c of the application of γ not being able to produce any trace not observed.

Proposition 5 *Suppose that if the SUT can produce trace $\bar{\sigma}x/y$ then the probability of receiving y in response to x after $\bar{\sigma}$ is at least p and that this holds for all x, y, and $\bar{\sigma}$. Let Γ' denote a set of adaptive test cases and each adaptive test case $\gamma' \in \Gamma'$ has been applied at least the following number of times:*

$$\frac{\log \delta}{\log(1 - p^{length(\gamma')})}$$

Suppose that O denotes the resultant traces, $det_f(\gamma, O)$ holds and let $E = Pre(IO(\gamma)) \setminus Pre(O)$. Let E' denote the minimal elements of E: $\bar{\sigma} \in E$ is in E' if no proper prefix of $\bar{\sigma}$ is in E. If $\delta \leq \frac{1-c}{|E'|}$ then we have confidence of at least c in the SUT not being able to produce any response to γ that has not been observed.

Proof. The SUT cannot produce any trace in E' if and only if it cannot produce any trace in E. From Proposition 4 we need $|E'|\delta \leq 1 - c$ in order to have confidence c as required. □

This result gives a condition under which a tester can have a given confidence in the SUT not being able to produce any response to γ that has not already been observed. If this confidence is set sufficiently high then the tester could choose not to apply γ if this condition holds.

5 The general case

In the general case we get the following result.

Proposition 6 *The set O of traces is guaranteed to contain the set of possible responses of the SUT to γ if and only if $IO(\gamma) \subseteq Pre(O)$.*

BURA

Proof. First, if $IO(\gamma) \subseteq Pre(O)$ then O contains every possible response of any SUT to γ and thus that we have seen all possible responses of the SUT to γ. Now suppose that the set O is guaranteed to contain the set of possible responses of the SUT to γ and $\bar{x}/\bar{y} \in Pre(IO(\gamma))$. We now observe that there exist implementations that produce $\bar{y}$ in response to $\bar{x}$ and thus we must have that $\bar{x}/\bar{y} \in Pre(O)$ as required. □

The following defines a function det such that $det(\gamma, O)$ returns true if and only if $IO(\gamma) \subseteq Pre(O)$.

Definition 4

$$
\begin{aligned}
det^1(\bar{\sigma}, null, O_{\bar{\sigma}}) &= true \\
det^1(\bar{\sigma}, (x, f), O_{\bar{\sigma}}) &= \forall y \in Y. det^1(\bar{\sigma}x/y, f(y), O_{\bar{\sigma}x/y}) \\
det(\gamma, O) &= det^1(\epsilon, \gamma, O)
\end{aligned}
$$

The proof of the following is similar to that for Proposition 2.

Proposition 7 *Given adaptive test case γ and a set O of traces, $det(\gamma, O)$ can be computed in time of $O(|Y||O|length(\gamma))$.*

6 Conclusions

In testing we observe traces of the SUT. If we can deduce that all traces that the SUT can produce in response to an adaptive test case γ have already been observed then we know that we do not have to apply γ. If this is the case then we can reduce the cost of test execution.

Previous work has considered the case when the SUT is known to be deterministic and we extended this by investigating the situation in which the SUT could be nondeterministic. We started with the case where we can apply a fairness assumption. We then weakened this assumption to there being a lower bound p on the probability of observing alternative responses of the SUT to any input and in any state. We then considered the general case. In all three cases we showed how the problem can be solved in low order polynomial time.

Since it is sometimes possible to deduce that all possible responses to an adaptive test case γ have already been observed in testing, the expected cost of testing depends upon the order in which we apply the adaptive test cases. This leads to the question of how we can minimise the expected cost of testing. This problem has been considered for a deterministic SUT and there are algorithms for producing preset orders [4] and for applying the adaptive test cases on the fly [7]. Future work will investigate how these approaches can be extended to a nondeterministic SUT.

References

[1] H. AboElFotoh, O. Abou-Rabia, and H. Ural. A test generation algorithm for protocols modeled as non–deterministic FSMs. *The Software Engineering Journal*, 8(4):184–188, 1993.

[2] J. Grabowski, A. Wiles, C. Willcock, and D. Hogrefe. On the design of the new testing language TTCN–3. In *Testing of Communicating Systems*, pages 161–176, Ottawa, August 29 – September 1 2000. Kluwer Academic Publishers.

[3] R. M. Hierons. Testing from a non–deterministic finite state machine using adaptive state counting. *IEEE Transactions on Computers*, 53(10):1330–1342, 2004.

[4] R. M. Hierons and H. Ural. Concerning the ordering of adaptive test sequences. In *23rd IFIP International Conference on Formal Techniques for Networked and Distributed Systems (FORTE 2003)*, volume 2767 of *Lecture Notes in Computer Science*, pages 289–302, Berlin, Germany, September 29 – October 2 2003. Springer–Verlag.

[5] Joint Technical Committee ISO/IEC JTC 1. *International Standard ISO/IEC 9646-1. Information Technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts.* ISO/IEC, 1994.

[6] ITU-T. *Recommendation Z.500 Framework on formal methods in conformance testing.* International Telecommunications Union, Geneva, Switzerland, 1997.

[7] G.-V. Jourdan, H. Ural, and N. Zaguia. Minimizing the number of inputs while applying adaptive tests. *Information Processing Letters*, 94(4):165–169, 2005.

[8] G. L. Luo, G. v. Bochmann, and A. Petrenko. Test selection based on communicating nondeterministic finite–state machines using a generalized Wp–method. *IEEE Transactions on Software Engineering*, 20(2):149–161, 1994.

[9] P. Tripathy and K. Naik. Generation of adaptive test cases from non–deterministic finite state models. In *Proceedings of the 5th International Workshop on Protocol Test Systems*, pages 309–320, Montreal, September 1992.

[10] H. Zhu and X. He. A methodology of testing high-level petri nets. *Information and Software Technology*, 44:473–489, 2002.

BURA